

On the Critical Behaviour, Crossover Point and Complexity of the Exact Cover Problem

Robin D. Morris¹, Vadim N. Smelyanskiy², and Daniel Shumow³

¹ RIACS, NASA Ames Research Center, MS 269-2,
Moffett Field, CA 94035-1000, USA
rdm@email.arc.nasa.gov

² NASA Ames Research Center, MS 269-2, Moffett Field, CA 94035-1000, USA
Vadim.N.Smelyanskiy@nasa.gov

³ University of Wisconsin-Madison

Abstract. Research into quantum algorithms for NP-complete problems has rekindled interest in the detailed study a broad class of combinatorial problems. A recent paper [1] applied the quantum adiabatic evolution algorithm to the *Exact Cover problem for 3-sets* (EC3), and provided an empirical evidence that the algorithm was polynomial. In this paper we provide a detailed study of the characteristics of the exact cover problem. We present the annealing approximation applied to EC3, which gives an over-estimate of the phase transition point. We also identify empirically the phase transition point. We also study the complexity of two classical algorithms on this problem: Davis-Putnam and Simulated Annealing. For these algorithms, EC3 is significantly easier than 3-SAT.

1 Introduction

In the early 1990s it was realized that many problems in the class of NP-complete problems [2] exhibit phase transitions, and that the peak in the complexity of practical algorithms occurs at the phase transition [3–5]. The example used in most of these studies is the *3-sat* problem, and extensive studies of the behaviour of 3-sat have been performed [6, 7]. Recently, however, there has been an upsurge in interest in other problems in the NP-complete class. This has been due to the rise in interest in quantum computation. Three problems that have been approached using the Quantum Adiabatic Evolution Algorithm (QAA) [8] are, the *exact cover problem for 3-sets* (EC3) [1], the *set partition problem* [9], and 3-sat [10]. In this paper we present the annealing approximation analysis of EC3, and the behaviour of two classical algorithms (the Davis-Putnam (DP) algorithm [11] and Simulated Annealing (SA)) applied to it. We present results that identify strongly the region of the phase transition.

Because DP has a very small factor in the exponent (see section 3.4), studying the asymptotic scaling of the complexity requires $N \gg 20$, the maximum value used in [1]. However, because the inner workings of QAA are much closer to SA than to DP [9, 12], we also study the complexity of SA, to try to determine

at what value of N the asymptotic regime is achieved. The complexity of SA scales with problem size⁴ as $\exp(0.023 \times N)$. This does not give much support to determining asymptotic exponential complexity from the empirical data that correspond to $N < 20$.

An N -bit instance of EC3 is built up from clauses, each of which is a constraint imposed on the values of three of the bits. If a given clause involves the three bits labeled i, j and k , then the constraint is satisfied if $z_i + z_j + z_k = 1$, ($z_i \in \{0, 1\}$) and is violated otherwise. There are 8 possible assignments for the values of z_i , z_j , and z_k , but only 3 out of the 8 satisfy the constraint. It is clear that satisfying assignments have one of three bits with value 1, and the remaining two with value 0. An N -bit instance of EC3 is a list of triples (i, j, k) (with distinct indexes in each triple) indicating which groups of three bits are involved in the clauses. The decision problem is to determine whether or not there is some assignment of the N bit values that satisfies all of the clauses. In general, there may not be any assignment that satisfy all the clauses and an optimization variant of EC3 is to find an assignment that violates the least number of clauses (in what following we shall refer to both variants of the problem simply as EC3, instead of using more conventional notation MAX-EC3 for optimization variant of this NP-complete problem).

2 Annealed Approximation Theory

The cost function $E_{\mathbf{z}}$ in EC3 is defined on a set of 2^N possible N -bit strings $\mathbf{z} = \{z_1, z_2, \dots, z_N\}$. It is equal to the number of constraints that are violated by a given assignment \mathbf{z} . We introduce an $M \times N$ matrix \mathcal{A} with M rows \mathbf{A}_m , one for each clause, and N columns, one for each bit in the N -bit string. Each row has three elements equal to 1, with the rest equal to 0. The elements of the m -th row $A_{m,l} = 1$ occur at distinct positions $l = i, j, k$ that coincide with the triple of indices in the corresponding clause. Then a given clause \mathbf{A}_m is satisfied by an assignment \mathbf{z} iff $\mathbf{A}_m \cdot \mathbf{z} = 1$, and the cost function can be written in the following form:

$$E_{\mathbf{z}} = M - \sum_{m=1}^M \Delta(\mathbf{A}_m \cdot \mathbf{z}, 1), \quad \mathbf{A}_m \cdot \mathbf{z} \equiv \sum_{l=1}^N A_{m,l} z_l. \quad (1)$$

Here M is the total number of clauses and $\Delta(k, l)$ is a Kronecker delta. The solution of EC3 is an assignment with minimal $E_{\mathbf{z}}$. In what follows we shall refer to a given instance of EC3 simply as \mathcal{A} .

Consider a set \mathcal{I} of all instances of EC3 with N bits and M distinct clauses. Due to the fact that the size of \mathcal{I} can be very large (equal to $\binom{M_{\max}}{M}$), where

⁴ Note that in a random problem instance, some of the bits are not included in any clauses. This differs from how the problem instances were generated in [1] and results in a smaller effective N (by a factor of ≈ 1.18)

$M_{\max} = \binom{N}{3}$) it is convenient to introduce the statistical distribution of instances within \mathcal{I} and sample from this distribution while studying the instance-dependent quantities such as algorithm complexity, number of solutions, etc. In this uniform sampling model with statistically independent clauses, one randomly chooses a triple of distinct indexes (i, j, k) from the interval $(1, n)$ and independently repeats this procedure M times - once for each row of \mathcal{A} .

2.1 Energy Distribution Function and Phase Transition in EC3 in the Annealing Approximation

The simplest energy distribution characterizing a given instance of EC3 is described by the following function:

$$P(E; \mathcal{A}) = 2^{-N} \sum_{\mathbf{z} \in \{0,1\}^N} \Delta[E_{\mathbf{z}}, E]. \quad (2)$$

Here $P(E; \mathcal{A})$ is the fraction of all possible assignments of \mathbf{z} that violate exactly E clauses ($E = 0, 1, \dots, M$) of an instance \mathcal{A} . Qualitatively the behavior of the function $P(E; \mathcal{A})$ can be analyzed by averaging it over the problem instances: $\bar{P}(E) = \langle P(E; \mathcal{A}) \rangle_{\mathcal{A}}$. To implement this we introduce the probability $p_{\mathbf{z}}$ that a single randomly generated clause is satisfied by a given assignment of \mathbf{z} :

$$p_{\mathbf{z}} = b(h(\mathbf{z})), \quad h(\mathbf{z}) = \sum_{j=1}^N z_j, \quad b(Q) = Q \binom{N-Q}{2} \binom{N}{3}^{-1}. \quad (3)$$

Note that this probability in EC3 depends only on the Hamming weight of a string $h(\mathbf{z})$ (i.e. the number of unit bits in it). With the assumption that clauses are independent, the ensemble-averaged distribution $\bar{P}(E) = \langle P(E; \mathcal{A}) \rangle_{\mathcal{A}}$ is given by a sum of binomial distributions (each corresponding to a certain value of the Hamming weight Q)

$$\bar{P}(E) = 2^{-N} \sum_{Q=0}^N \binom{N}{Q} \bar{P}(E, Q), \quad \bar{P}(E, Q) \equiv \binom{M}{E} b^{M-E}(Q) (1 - b(Q))^E. \quad (4)$$

($E = 0, 1, \dots, M$). Direct averaging of quantities like (2) over the ensemble of \mathcal{A} is usually called in spin-glass theory an "annealing" approximation [13].

For a given problem instance the number of solutions always has a multiplicative factor equal to $2^{\alpha N}$ where α is the fraction of bits that are not included in any clauses. The average value of α is

$$\bar{\alpha} = \binom{N-1}{3}^M \binom{N}{3}^{-M}, \quad \bar{\alpha} \rightarrow \exp(-3\gamma) \quad \gamma = \frac{M}{N} \quad (N \rightarrow \infty), \quad (5)$$

and it decreases exponentially with the ratio $\gamma = M/N$.

Consider now the average number of bit configurations that violate exactly E clauses, $2^N \bar{P}(E)$, normalized by the multiplicative factor $2^{\alpha N}$

$$\bar{\Gamma}(E) = 2^{N-\alpha N} \bar{P}(E). \quad (6)$$

For small E values $E \ll M^{1/2}$ and in a certain range of γ the function $\bar{\Gamma}(E)$ takes the following form:

$$\bar{\Gamma}(E) \approx \frac{N^E}{E!} [\bar{\gamma}_c (\zeta - 1)]^E \zeta^{-N(\gamma - \bar{\gamma}_c)}, \quad |\gamma - \bar{\gamma}_c| \ll 1. \quad (7)$$

Here $\gamma = M/N$, the constants $\bar{\gamma}_c \approx 0.703$, and $\zeta \approx 2.29$. Eq. (7) describes the exponential growth of the number of assignments with E at small E .

We will compare this annealing approximation to the critical value ($\bar{\gamma}_c \approx 0.603$) with the experimental value to be determined later (section 3.3).

We note that if one fixes the value $\gamma < \gamma_c$ in (7) and considers the asymptotic limit $N \rightarrow \infty$ then the ensemble-average fraction of normalized number of solutions $\bar{\Gamma}(0)$ will grow exponentially with N . However if $\gamma > \gamma_c$ then $\bar{\Gamma}(0)$ is exponentially decreasing with N . Therefore Eq. (7) describes a phase transition in the number of satisfying assignments, $E_{\text{av}} = 0$, between the “over-constrained” ($\gamma > \bar{\gamma}_c$) and under-constrained ($\gamma < \bar{\gamma}_c$) regimes of EC3 as a control parameter $\gamma = M/N$ varies through the critical region $\gamma \approx \bar{\gamma}_c$.

The average *minimum* energy $E_{\min} = 0$ for $\gamma < \gamma_c$, and for $\gamma > \gamma_c$ it satisfies the condition $\bar{\Gamma}(E_{\min}) = 1$. From this one can obtain:

$$E_{\min} = N(\gamma - \gamma_c) \log \zeta / \log \left(\frac{1}{(\gamma - \gamma_c) \log \zeta} \right), \quad (\gamma > \gamma_c). \quad (8)$$

Therefore in the over-constrained range $\gamma > \gamma_c$ the minimum energy grows linearly with N (see section 3.6 for a comparison with the experimental results). Comparison between the analytical result $\bar{P}(E)$ (4) and simulations of the exact function $P(E; \mathcal{A})$ computed for randomly generated problem instances is shown in figure 1. We note that more detailed analysis based on the method of replicas [13] reveals the deviation of $P(E; \mathcal{A})$ from the annealing approximation based on Eq.(4).

Because the probability for a randomly-generated clause to be satisfied by a given assignment depends only on the Hamming weight of this assignment it is of interest to consider the average number of bits in a satisfying assignment (the solution). By taking the sum in the (4) over Q with the method of steepest descent, the dominant contribution comes from the small vicinity of the value $Q = Q^*$, which is a stationary point of the integrand in (4). After setting $E = 0$ we obtain the stationary point equation in the following form

$$x^*(1 - x^*) \log \frac{1 - x^*}{x^*} - 2\gamma x^* + \gamma = 0, \quad x^* = \frac{Q^*}{N}, \quad \gamma = \frac{M}{N}. \quad (9)$$

Its solution for the critical value of $\gamma = \gamma_c$ is $x^* \approx 0.3808\dots$ that gives the fraction of unit bits in the solution string. Experimentally, the fraction was 0.38.

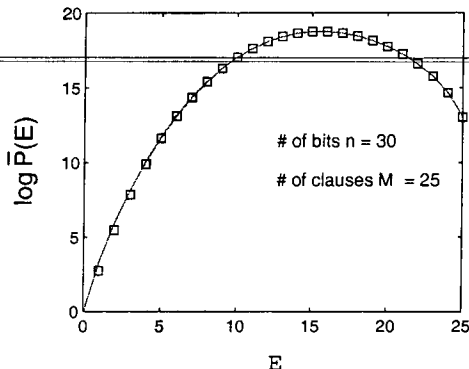


Fig. 1. Logarithmic plot of the distribution of the cost values in EC3: rectangles corresponds to numerical simulations of $P(E; \mathcal{A})$ vs E based on (2); solid curve corresponds to analytical expression (4) $\bar{P}(E)$ vs E in the annealing approximation.

We note that a true critical value for the ratio M/N at the phase transition point is different from that given in annealing approximation because the disorder effect is significant on the tail of $P(E; \mathcal{A})$ (for $E \rightarrow 0$) that determines that ratio. Detailed numerical study of the phase transition point is given below. We note in passing that near the maximum of $P(E; \mathcal{A})$ this function is well approximated by the approximate result (4). In particular it has a Gaussian form in this range of E , with the mean value and variance given below:

$$\langle E \rangle = E_{\max} = \frac{5\gamma}{8}N, \quad \langle E - \langle E \rangle \rangle^2 = \sigma^2(0)N, \quad \sigma^2(0) \approx \frac{3}{64}(3\gamma^2 + 5\gamma) + \mathcal{O}(1). \quad (10)$$

Here the expression for E_{\max} determines the value of cost that occurs most often if the assignments \mathbf{z} are sampled at random. This value is independent of particular instances of EC3. The normalized standard deviation $\sigma(0)$ is a self-averaging quantity given in (10). In the limit $N \rightarrow \infty$ the first and second moment correctly describe the form of the distribution near the maximum $|E - E_{\max}| \ll N$, which is not sufficient however for the analysis of the local search algorithms such as simulated annealing that spend most of the time for small values of $E \sim 1$.

3 Experimental Results

In this section we present experimental results on random EC3 instances. Using the Davis-Putnam (DP) algorithm and Simulated Annealing (SA), we study the crossover point, the computation complexity and the mean minimum energy of solution. We also identify experimentally the position of the phase transition.

Table 1. Outline of the Davis-Putnam Algorithm

```

Find_Model( theory )
  unit_propagate( theory );
  if contradiction discovered return(false);
  else if all variables are valued return(true);
  else {
    x = some unvalued variable;
    return( Find_Model( theory AND x ) OR
            Find_Model( theory AND NOT x ) );
  }

```

3.1 The Davis-Putnam Algorithm

The Davis-Putnam (DP) algorithm [11], or a variation, is regarded as the most efficient *complete* algorithm for satisfiability problems. An outline of the DP algorithm is given in table 1 [6]. The version we used varies from this outline in one major respect. We perform a sort of the variables before the first call to `Find_Model`, sorting on the number of clauses which use the variable. This was found to produce, on average, a very large speed-up in the algorithm's execution.

The `unit_propagate` step of the algorithm is also extremely efficient for the exact cover problem. Once one variable in a clause is set to 1, the value of the other two variables is fixed, and extensive propagation often occurs. Also, because a single variable in a clause being set to 1 determines the other two variables in the clause, we call `Find_Model(theory AND x)` first.

3.2 The Formation of a Super Cluster

A basic characteristic of problems that exhibit a phase transition is the formation of a super-cluster [14]. If the constraints are used to cluster the variables into groups that are (indirectly) connected, then once the number of constraints reaches a critical density, a "super-cluster" forms that includes the vast majority of the variables that are involved in the clauses. In Figure 2 we plot the ratio of the size of the second largest cluster to the size of the largest cluster, for increasing N . The phase transition is clearly seen.

3.3 The Crossover Point

The major feature of a phase transition in a satisfiability problem is the presence of a threshold in γ , below which almost all random problem instances are solvable, and above which almost no random problem instances are. Figure 3 shows a plot of the proportion of random problem instances that have a satisfying assignment, versus γ , for various values of N . The proportions are based on running the DP algorithm on 50,000 random problem instances for each value of N and γ . The expected features are present. The sharpness of the phase transition

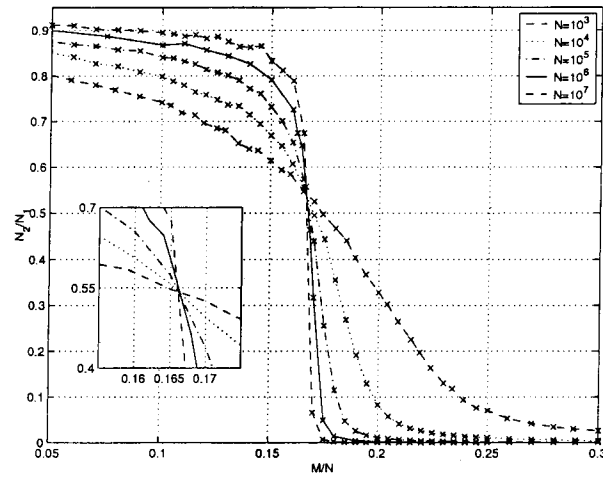


Fig. 2. Ratio of the size of the second largest cluster to the size of the largest cluster

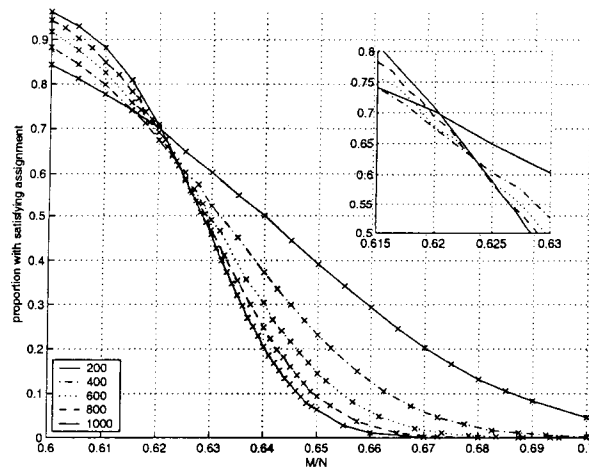


Fig. 3. Proportion of problem instances with a satisfying assignment

increases with N , and the point at which the curve crosses the line where the proportion of instances with a satisfying assignment equals 0.5 decreases with N .

Experimentally the crossover point is at $\gamma_c \approx 0.625$. Note that this is somewhat lower than $\bar{\gamma}_c = 0.703$ given by the annealing approximation, showing that the tails of $P(E; A)$ are important. In figure 4 (lower curve) we plot the value of γ for which 50% of the problem instances were satisfiable as a function of the number of bits. The curve appears to have an asymptote around $\gamma_c \approx 0.625$.

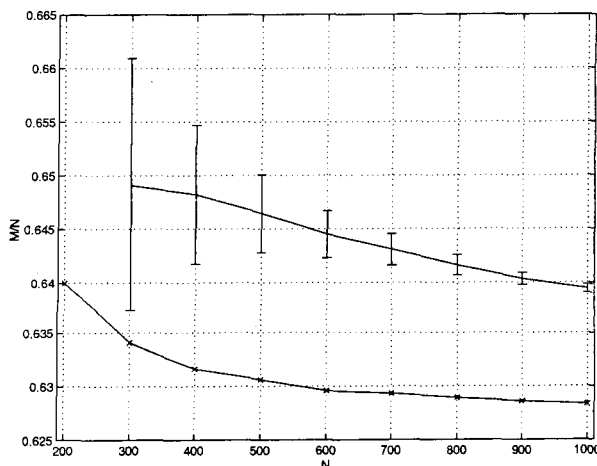


Fig. 4. Top curve: plot of the maximum complexity of the DP algorithm. Lower curve: the position of the crossing point of proportion with satisfying assignment = 0.5

3.4 Complexity of the Davis-Putnam Algorithm

Figure 5 shows plots of the median complexity of the Davis-Putnam (DP) algorithm, where complexity is defined as the number of calls to `Find_Model` (Table 1). The median was taken over 50,000 random problem instances. As expected, because the DP algorithm is complete, its performance scales exponentially with problem size, N . Note also that the value of γ for which the maximum complexity occurs is above γ_c , and slowly reduces as N increases. In figure 4 (upper curve) we plot the position of the maximum complexity and its uncertainty. we note that for the range of values of N considered, it does not appear to have converged to an asymptote, but the curve does not appear to contradict our earlier result of $\gamma_c \approx 0.625$.

Fitting an exponential law to the peak complexity gives $C = 6.13 \exp(0.0067 \times N)$, a very slow rate of increase – an order of magnitude slower than reported results on the complexity of DP applied to 3-SAT [6].

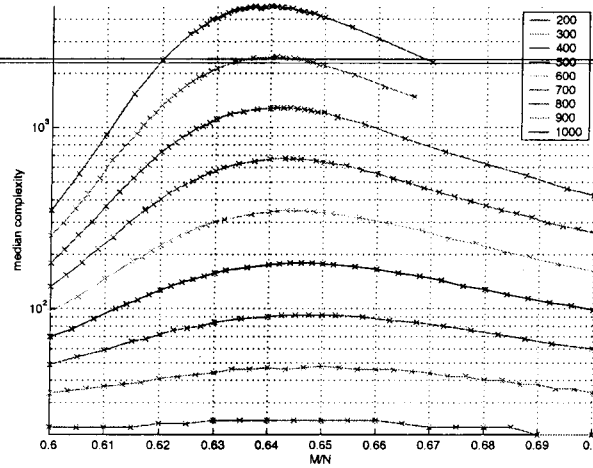


Fig. 5. Computational complexity of DP (all problem instances)

3.5 Complexity of Simulated Annealing

In order to quantify the complexity of Simulated Annealing (SA), it is necessary to be able to determine when the minimum energy configuration has been reached. In order to be able to do this, when a random problem instance was generated, we first used DP to determine if it had a zero-cost solution. If so, then SA was applied; if not, the problem instance was rejected and a new random problem instance was generated, etc.

We used an exponential cooling schedule, $T = T_0 a^k$, where an iteration k is one attempted flip of all N bits. For each problem instance, we set $T_0 = 2$, ran a range of values of a , and ran the SA algorithm 100 times. We define complexity as follows. For those runs that found the zero-cost solution, compute the mean number of iterations⁵. The complexity is defined as this mean number of iterations, divided by the proportion of the 100 runs that found a zero-cost solution. In figure 6 we plot the minimum complexity over the values of the cooling schedule, a . The flattening off of the curves for low values of γ is due to the lowest values of a used being too high in this region.

Note that the scaling behaviour is very similar to that for the DP algorithm, but that the rate of increase of complexity with N is much larger ($C \propto \exp(0.023 \times N)$).

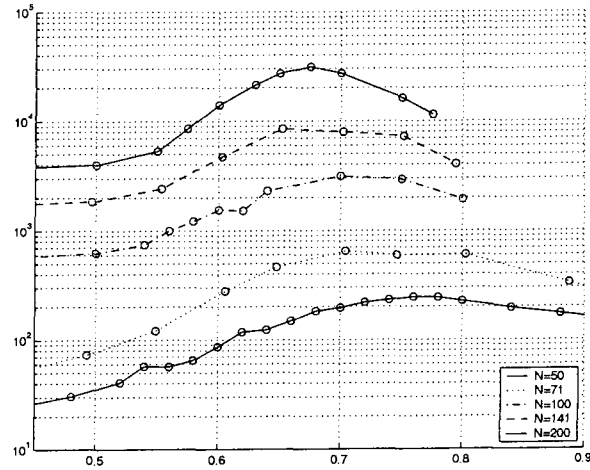


Fig. 6. Computational complexity of simulated annealing (problems known to have a solution)

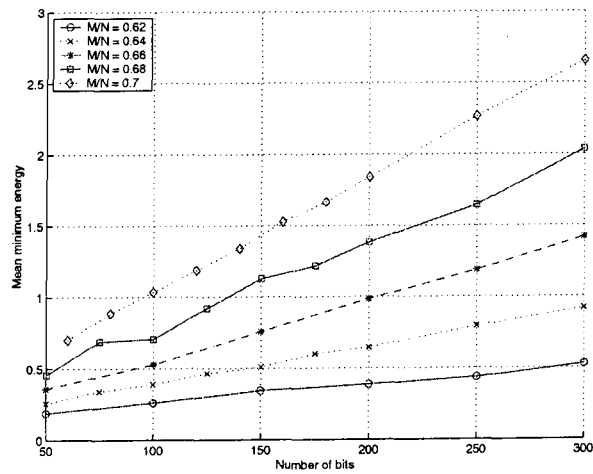


Fig. 7. The mean of the minimum energy state as a function of N , for values of M/N just above the phase transition

3.6 Scaling of the Minimum Energy Close to the Phase Transition

We also used SA to obtain the minimum energy for random EC3 instances close to the phase transition. Figure 7 plots the mean (over 800 instances) minimum energy versus N for values of γ near the phase transition. These graphs qualitatively support the scaling behaviour given by the annealing approximation in section 2 (see eq. (8)).

4 Conclusions

We have presented theoretical and experimental results on the algorithmic complexity and position of the phase transition for the EC3 problem. The annealing approximation gives $\bar{\gamma}_c = 0.703$ and experiments suggest $\gamma_c \approx 0.625$. Unpublished results [15] give analytic bounds of $0.517 < \gamma_c < 0.727$ and are in good agreement with our theoretical and experimental values.

In passing we want to note that a recent paper [10] considered simulations of QAA for the 3SAT problem and its comparison with the results of classical search algorithms, including GSAT, at the phase transition point. The exponential fit to GSAT was $\exp(0.13 \times N)$ which was about the same as for QAA for that problem. Recall that SA for EC3 has complexity $\exp(0.023 \times N)$. We studied the complexity of the simplest local search method for global optimization, SA, because the dynamics of this algorithm can be viewed as the closest classical counterpart to QAA, and may cast light on the difficulty of solving EC3 using QAA.

Acknowledgments

One of us (VNS) acknowledges the support of the NASA CICT/Revolutionary Computing Algorithms program (project N0: 749-40).

References

1. Farhi, E., Goldstone, J., Gutmann, S., Lapan, J., Lundgren, A. and Preda, D.: A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem *Science*, **292**, pp 472–475, (2002)
2. Garey, M.R. and Johnson, D.S.: *Computers and Intractability: A Guide to the Theory of Np-Completeness*. W H Freeman & Co. (1979)
3. Cheeseman, P., Kanefsky, B. and Taylor, W.M. Where the Really Hard Problems Are. *Proc. Twelfth International Joint Conference on Artificial Intelligence, IJCAI-91*, Sidney, Australia (1991)

⁵ Note that for those runs that did **not** find a zero-cost solution, the minimum cost found was usually first achieved in **approximately** the same number of iterations as needed to find the zero-cost solution. This demonstrates clearly that SA is essentially a local search method.

4. Mitchell, D., Selman, B. and Levesque, H.: Hard and Easy Distributions of SAT Problems. Proceedings of the Tenth National Conference on Artificial Intelligence (1992)
5. Kirkpatrick, S. and Selman, B.: Critical Behavior in the satisfiability of random Boolean expressions. *Science*, 264, pp 1297–1301, (1994)
6. Crawford, J.M. and Auton, L.D.: Experimental Results on the Crossover Point in Satisfiability Problems. Proc. Eleventh National Conference on Artificial Intelligence (1993)
7. Crawford, J.M. and Auton, L.D.: Experimental results on the crossover point in random 3-SAT. *Artificial Intelligence*, 81(1):31–57, (1996)
8. Farhi, E., Goldstone, J., Gutmann, S. and Sipser, M.: Quantum Computation by adiabatic evolution. [arXiv:quant-ph/0001106](https://arxiv.org/abs/quant-ph/0001106)
9. Smelyanskiy, V.N, Toussaint, U.V. and Timucin, D.A.: Dynamics of quantum adiabatic evolution algorithm for Number Partitioning. [arXiv:quant-ph/0202155](https://arxiv.org/abs/quant-ph/0202155)
10. Hogg, T.: Solving random satisfiability problems with quantum computers. [arXiv:quant-ph/0104048](https://arxiv.org/abs/quant-ph/0104048)
11. Davis, M. and Putnam, H.: A computing procedure for quantification theory. *J. ACM*, 7:201-215 (1960)
12. van Dam, W., Miska, M. and Vazirani, U.: How powerful is Adiabatic Quantum Computation? [arXiv:quant-ph/0206003](https://arxiv.org/abs/quant-ph/0206003)
13. Mezard, M., Parizi, G. and Virasoro, M.: Spin glass theory and beyond. World Scientific, Singapore, 1987.
14. Erdős, P. and Rényi, A.: Publication of the Mathematics Institute of the Hungarian Academy of Sciences, 5:17, (1960)
15. Boufkhad, Y., Kalapala, V. and Moore, C.: Phase Transition in Positive 1-in-3 SAT. Private communication (C. Moore), (2003)
16. Farhi, E., Goldstone, J. and Gutmann, S.: Quantum Adiabatic Evolution Algorithms versus Simulated Annealing. [arXiv:quant-ph/0201031](https://arxiv.org/abs/quant-ph/0201031)